The Type



简体中文 | 繁體中文

参数化设计与字体战争：从 OpenType 1.8 说起

谭 沛然 | 2016/09/28

2016 年 9 月 14 日，ATypI （Association Typographique Internationale, 国际文字设计协会）第 60 届年会在波兰华沙召开。会上，微软、Adobe、苹果和谷歌联合发布了第 1.8 版 OpenType 规范，宣布支持用户端无级变换的功能，命名为 OpenType Variable Font（OpenType 可变字体）。

字体界无不震动。Dalton Maag、Dutch Type Library、Font Bureau、蒙纳、森泽和华文等多家字体厂商发来祝贺，两大字体设计软件 FontLab 和 Glyphs 率先宣布支持新规范。推特上的平面和字体设计师欣喜若狂，奔走相告。字体设计师 John Hudson （约翰·哈德森）宣称：「OpenType 可变字体，是数码字体近十年来最伟大的进展。」

字体的无级可变性，早已不是新鲜话题。但 OpenType 1.8 能够走到今天这一步，背后经历的风雨不可谓不漫长。现在，我希望带读者回到四十年前，从数码字体的开端说起。

1. 背景

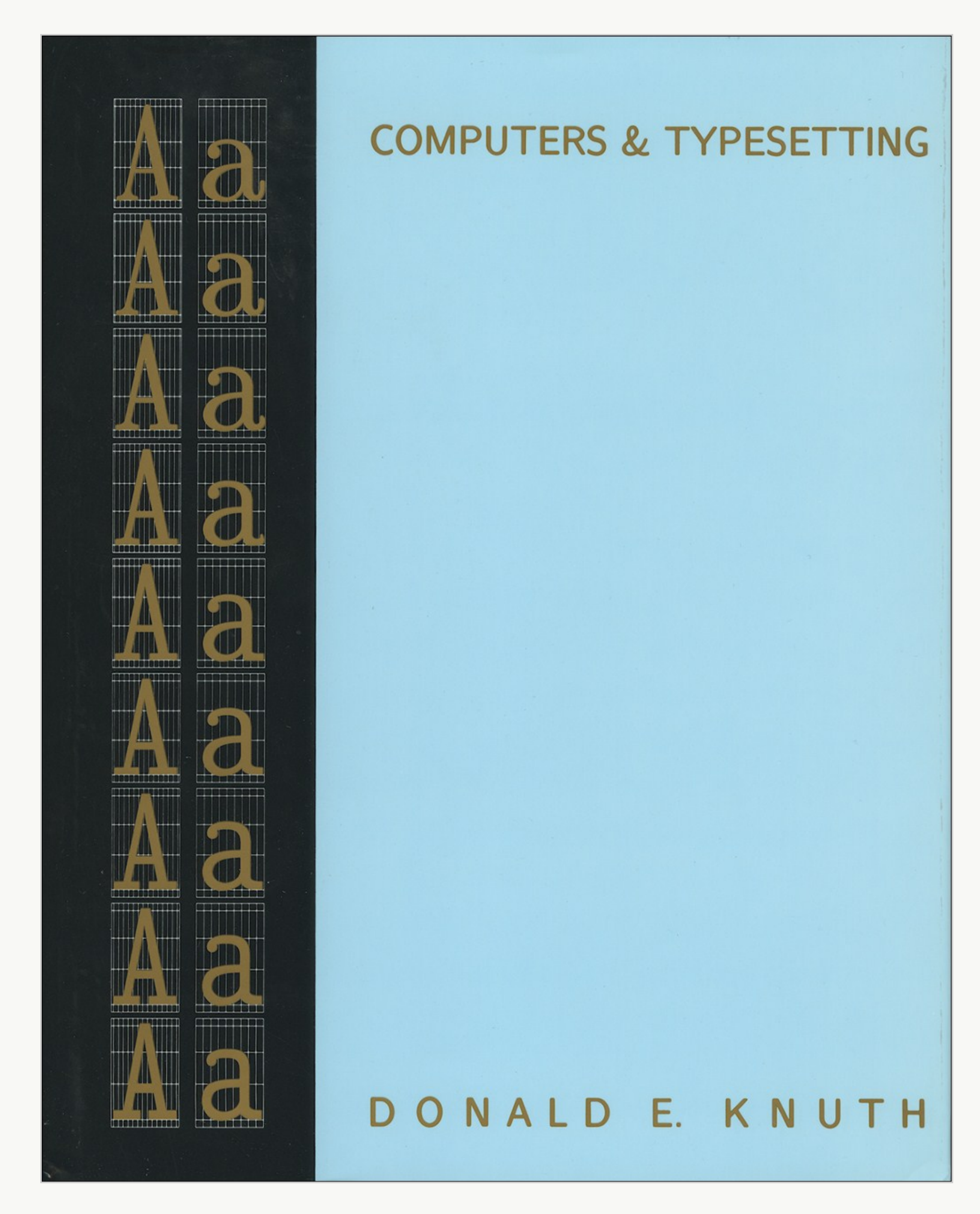
1970 年代，西方的字体业界正在从照相排版 (Photocomposition) 转移到数码排版 (Digital Publishing)。彼时技术非常不成熟，无论是行将就木的照排机，还是方兴未艾的电脑，印刷效果都非常差。

对编程技术有所涉猎的读者，都可能知道高德纳 (Donald Knuth, 1938—) 的鼎鼎大名。那时，他的巨著《计算机编程艺术》(*The Art of Computer Programming*) 正准备出版第四卷。但当他看到泛滥的、丑陋的字体和排版时，他觉得自己的书绝不能排成这样。于是他自己动手，在1979年写出了一套字体设计软件，称为 Metafont，并模仿着蒙纳 8a 的铅活字，用 Metafont 设计了 Computer Modern 系列字体。他后来又创作出 TeX 排版软件，此处按下不提。

Metafont 的理念，极为高瞻远瞩。当下流行的字体设计软件，都是以编辑轮廓 (outline) 为基础。但高德纳认为，Metafont 的目的是要教会电脑如何绘画字形，因此他把「笔」的概念带入了设计中。在 Metafont 里，电脑似乎是握着一支可以变换形状的笔，在给定的单元网格内「写」出需要的字母，而这「写」的过程就可以用代码来控制。

而由于高德纳把轮廓的绘制还原成笔划的结果，对这支「笔」的定制也变得越来越复杂。而这种对「笔」的描述和定制，便打开了参数化字体设计 (Parametric type design) 的大门。只要参数足够多、足够精细，从衬线的长

短、弧度，到 x 字高的比例、字母的宽窄……任何一个字母的形状都可以任意地、无级地变换。实际上，高德纳自己设计的 Computer Modern 字体家族，本身就带有若干种风格不同的变体，全部都是通过调整参数得来。 [1]

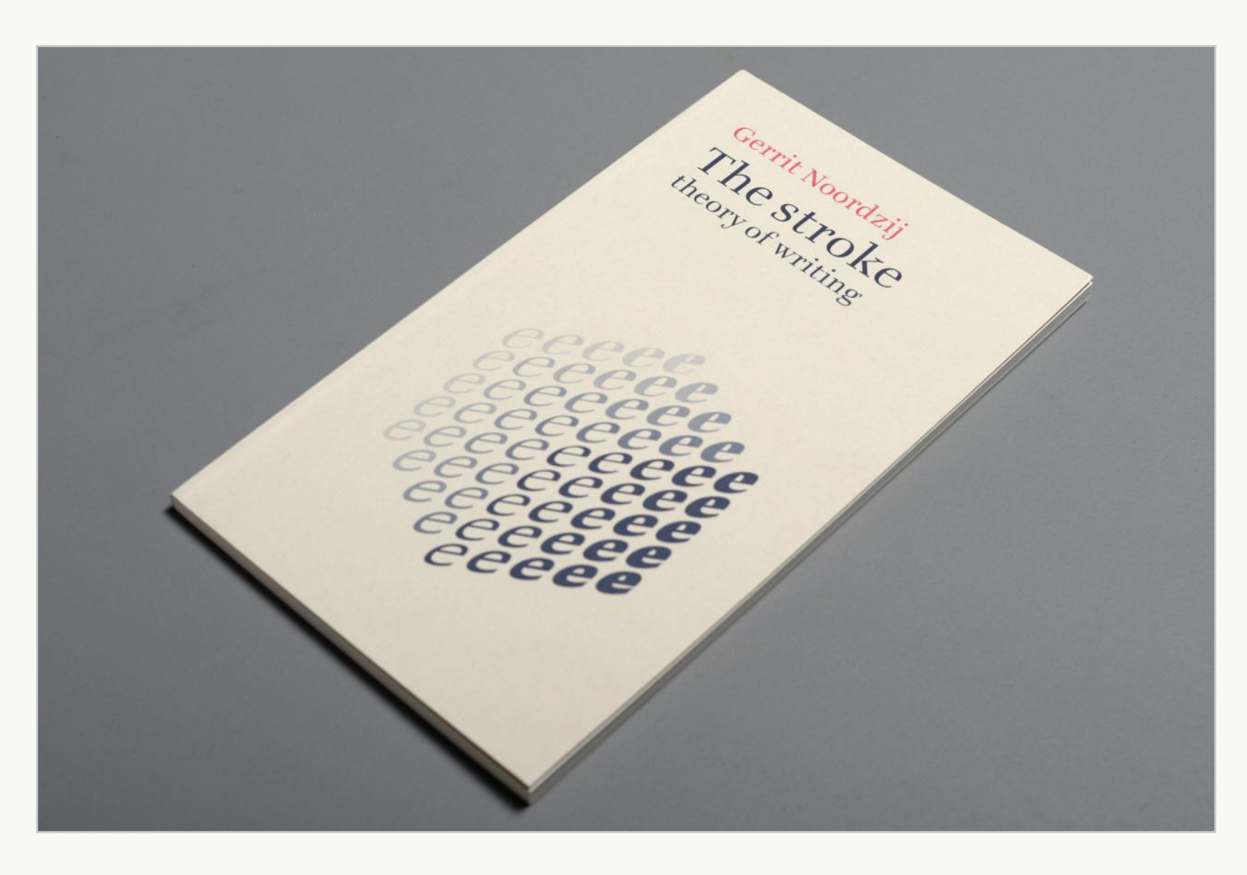


— 高德纳《电脑与排版》封面。其中从上到下的不同风格的小写 a 都是通过 Metafont 变换 Computer Modern 生成的。图：informit

Metafont 另外一个重要的意义，是对字型本身的理解。基于轮廓本身的字体设计软件，把字形当成图形来看待；而 Metafont 由于对每个字形都有独

立的控制代码，实际上是把它们都当成一个个独立的小程序来看待。

这有好的一面，也有坏的一面。好的一面是，这样的高瞻远瞩，最终会在十年后被微软和苹果实现；而坏的一面是，这使得用 Metafont 做字体设计非常困难。字体设计师主要依靠的是视觉的结果，而不能即时预览、纯粹靠改动数值来使用的编辑器是非常难使用的。这直接导致 Metafont 没能成为主流的字体设计软件，而它里面深刻的思想、精良的曲线自动调整技术，也随之被掩埋。[2]



— *The Stroke: Theory of Writing* 英文版，由 Peter Enneson 在1995年译出。图：Typotheque

无独有偶，在地球的另一面，也有人提出了相似的理念。他的名字是 Gerrit Noordzij（赫里特·诺尔泽，1931— ），荷兰书法家、碑刻家、平面设计师和字体设计师。根据他常年研究书法和碑刻的成果，他在 1985 年出版了《笔划：书写的理论》(*The Stroke: Theory of Writing*) 一书。此书虽小，但影响巨大。Noordzij 以此建立起一套字体设计的理论体系，并用这套体系建立起当代字体设计中的一个独立流派，以海牙皇家美院 (KABK, Koninklijke Academie van Beeldende Kunsten) 为据点，上至 Frank E. Blokland（弗兰克· E. 布洛克兰德），下到 Nina Stössinger （尼娜·施特辛格），荷兰字体设计师很少不受其影响。



—图：Vimeo

跟高德纳一样，Noordzij 也不认为字形应该纯粹由轮廓描述，而应该是一支虚拟的笔书写出来的；不同的字形轮廓，是不同的笔以不同的方式书写的结果。这支笔可以是平头、定宽的，其结果就是「旧体」(Garalde / Old style)；可以是尖头、变宽的，其结果就是「过渡体」(Transitional)。对这支笔的描述，自然也就成为了字体设计的参数。其中 Noordzij 认为最重要的三种参数是：平移 (Translation)、扩张 (Expansion) 和转笔 (Rotation)。为了更好地表达这三种参数之间的相互关系，他还画出了一个立方体图形，称之为 Noordzij Cube。Noordzij Cube 的重要意义在于，它直观明确地展示出，不同字体风格之间并不是完全独立、各自分离的，而是互通的。只要调整底下在书写的笔，不同风格的不同字体，甚至可以互相渐变。Gill Sans 可以渐变成为 Baskerville，Garamond 也可以渐变成为 Frutiger。

虽然 Noordzij 的理论还存有争议，但有一点毋庸置疑：如果说高德纳在程序上提出了概念、证明了参数化的可行性，那么 Noordzij 则是从设计和艺术这一边，为参数化字体设计进一步铺平了道路。

2. 桌上出版革命

就在 Noordzij 发表《笔划》一书的三年前，1982 年 John Warnock（约翰·沃诺克）联合 Charles Geschke（查尔斯·杰什克）创建了 Adobe 公司。1984 年，Adobe 发表了 PostScript 页面描述语言。

那时，桌面排版 (Desktop Publishing) 的技术才刚刚开始发展，「所见即

所得」(WYSIWYG, What You See is What You Get) 对桌面排版而言仍然是高端技术，尚未平民化。要使用桌面电脑排版和印刷，用户的电脑里首先要存储有轮廓字体（outline font, 与「点阵字体」bitmap font 相对）文件。在没有「所见即所得」的情况下，所有的页面布局和设计只能靠一次又一次的猜测和试错。在用户经过反复推测、尝试，完成页面设计后，软件会用 PostScript 语言把这个页面记录下来，然后输送给打印机。在 PostScript 之前，打印机只能输出低精度的点阵，而且效率缓慢、要价昂贵。

这时候虽然已经有了廉价、可供个人购买的打印机，但要进行正规的印刷作业，还需要聘请专门的服务所 (Service Bureau)。因为只有高端的打印机才能输出令人看起来舒服的效果，而高端的打印机当时价格并不是个人能够承受得起，就出现了印刷服务所这一行业，而它同时也是古老的印刷工坊 (Printing Shop) 的某种延续。

在回归苹果后不久，热衷于图形艺术和书法的乔布斯参观了 Adobe 公司，并迅速意识到数码字体和桌上排版是一个未来的热门需求。他强烈推荐 Warnock 把 PostScript 推广到打印机上。在 Adobe 和苹果的打印机技术合作中，苹果出资 250 万美元，买入了 Adobe 公司 15% 的股票。

支持 PostScript 的打印机装有一个解读器程序 (PostScript interpreter)，把用户送来的 PostScript 文件转换为图像，打印机输出的实际上是一张很高分辨率的图片。这一解读又被称为栅格化 (Rasterization)，是把轮廓数据转换为图像数据的过程，因此解读器程序也被简称为栅格化引擎 (Rasterizer)。1985 年，苹果基于佳能的技术推出了自己的 LaserWriter 打印机，率先支持解读 PostScript 文件，这在推广 PostScript 和桌面排版扮演了重要角色。在这之前不久，佳能曾经与另一家地处加州的激光打印机公司 Imagen 合作过，但 Imagen 的打印机并没有获得像苹果 LaserWriter 那样的广泛成功，因此没能在历史上留下印记。

Adobe 不仅率先推广了一套桌上排版的印刷规范，同时还设计了与之相配套的字体文件格式，做到了产业链整合。Adobe 设计了若干种字体格式，其中最著名的就是 PostScript Type 1（PostScript 第一型），简称 PS Type 1。与 Metafont 记录一种虚拟的「书写」不同，Type 1 格式采用三次贝塞尔曲线 (cubic bezier curve)，记录的是字形的外轮廓。[3]

其他有名的字体格式包括 Type 2，基于 Type 1 改动而成，主要是为了与 CFF 相兼容[4]。配合 CFF，Type 2 被广泛应用到 PDF 嵌入字体中。此外，如果打印机的固件内部随附了字体，这些字体也是使用 Type 2 来存储轮廓曲线数据的。另外一种就是 Type 3，它是 Type 1 的改进版，支持字形内带颜色等

一系列高级功能，但是打印机的硬件支持非常少，所以不大常用。还有一种比较有趣的是 Type 42，它的名字来源则是《银河系漫游指南》中解释一切事物、终结所有问题的答案「42」[5]。

有了字体文件，还需要让用户去使用它们。在推出 LaserWriter 的同一年，苹果与当时的排版软件公司 Aldus 会谈（Aldus 之名，即来自文艺复兴时期意大利书法家阿尔杜斯·马努提乌斯 Aldus Manutius）。1985 年，Aldus PageMaker 排版软件推出苹果版本，支持使用 PostScript 格式的字体进行版面设计[6]，并可配合苹果 LaserWriter 进行打印。

最后，当然就是要有好的字体。为了更好地推广 PostScript，Adobe 创立了一个庞大的高质量字体库。有的字体是直接从著名字体公司购买的授权，比如 Times New Roman 和 Helvetica；而另一部分则是聘请设计师从新设计。这部分为了推广 PostScript 而设计的字体，其中就包括读者应已熟知的 Minion、Adobe Garamond、 Myriad 和 Garamond Premier 等等，它们都有一个共同的系列名：Adobe Originals （Adobe 原创系列）。这群设计师中，最著名的是一男一女：Robert Slimbach （罗伯特·斯林巴赫，主要字体作品有 Adobe Jenson、Adobe Garamond、Garamond Premier Pro、Warnock、Caflisch Script）和 Carol Twombly（卡罗·通布利，主要字体作品有 Chaparral、Trajan、Nueva）。



— Robert Slimbach 与 Carol Twombly。图：Tumblr

这样，Adobe 提供字体设计、字体使用、文档排版和打印机输出的一整套技术规范，Aldus 专注于排版设计软件的开发，而苹果则以 Mac 电脑为基础，把这些技术带给千千万万的普通民众。古老、缓慢、昂贵的活字和照相排版，正在被更为廉价、方便的计算机代替。桌上出版的燎原之火，从此也被点燃。

3. Royal

Adobe、苹果和 Aldus 联合推动桌上出版革命，都得到了重要的利益。Aldus 卖出了更多的软件，苹果则通过在设计师和艺术家中的广泛使用，扭转了 Mac 电脑销量的颓势，建立了在文艺工作者中的声望。而 Adobe 则从一个名不见经传的小公司，一举成为财力雄厚的大企业，掌控着数字出版流程中的关键基础。为了进一步巩固自己在市场上的垄断地位，Adobe 尽管公开了其他 PostScript 字体格式的技术规格，却对 Type 1 的规格一直保密。同时，所有使用 Type 1 规格来生产的字体，数据都被加密了。这也就意味着，字体厂商如果要使用 Type 1 这种主流格式来生产字体，就需要向 Adobe 支付高昂的授权费用，否则就只能使用 Type 3 这种没人支持的格式。这既让 Adobe 一时保证了字体和排版业界的垄断地位，也在业界渐渐催生了不满。

另一方面，乔布斯认识到，要进一步提升 Mac 电脑的销量、为广大普通用户带来便利，就要让电脑做到所见即所得。当时时，用户在电脑上进行排版设计是无法实时预览的。这是因为关键的技术，即解读 PostScript 并将它转换为图像的栅格化引擎安装在打印机上，而不是电脑上。只有把 PostScript 的轮廓数据转成像素数据，才能够在屏幕上显示。因此，乔布斯向 Warnock 提出，让 Adobe 把 PostScript 栅格化的技术授权给苹果，让 Mac 电脑也能自带解读 PostScript 语言的能力。

但 Warnock 拒绝了。他的理由是，如果把 PostScript 栅格化引擎给了苹果，那么 Adobe 就会失去当时的垄断地位，更别提苹果会在这之上做的种种修改了。一旦让别的公司发布特制版的 PostScript，它作为业界唯一权威规范的地位也会被威胁。另外一个重要的原因是，如果个人电脑拥有了从排版设计到输出图像文件的全部能力，那么打印机就不必再配备昂贵的处理器，也不需要从 Adobe 购买解读 PostScript 数据的程序，而仅仅需要处理图像。Adobe 当然也意识到所见即所得对于桌上出版的重要性，因此也开发了 Display

PostScript，以支持实时预览 Type 1 字体，并曾经尝试过把它卖给苹果和微软。然而 Adobe 索要的授权费用太高，交易最终没能达成。

在被 Adobe 拒绝后，苹果决心另立门户，自行开发一套技术，覆盖从字体设计、生产，到排版、输出的全流程，与 Adobe 抗争。为此，在 1987 年苹果转向了微软。两者搁置往日的商业争端，暂时握手言和。不久，苹果就和 Adobe 彻底撕破了脸皮。

在苹果和微软的合作中，苹果负责开发字体文件格式，覆盖设计和排版，这就是 TrueType；微软则负责开发与打印机的对接，也就是栅格化引擎，力图追赶、甚至替代 PostScript，被称为 TrueImage。在这样的合作中，苹果采取了与 Adobe 完全相反的做法，在开发完字体格式之后，把技术规范全部公开，让所有人都能使用这样的技术来设计字体，不收取一分钱的授权费（No royalty）。因此，这个计划的秘密代号就是 Royal。Royal 计划的主要目标之一就是在个人电脑的操作系统上内建栅格化引擎，让普通用户不仅能实时预览设计，还能直接输出到打印机，并与 PostScript 相容。

苹果自己的 TrueType 开发计划，则是一个更大的系统套件中的组成部分，这个系统套件叫做 GX，负责 Mac OS 的视觉呈现。所以，TrueType 的原名是 TrueType GX。另一个 GX 家族的成员，是图形渲染引擎 QuickDraw GX。

为了执行 Royal 计划，苹果招募了一批既懂技术，又懂文字设计的工程师，而他们之中的大部分，都来自上文提到的、名不见经传的打印机制造商 Imagen。主要队员包括：

- Sampo Kaasila（桑波·卡西拉），前 Imagen 工程师，提出了大部分 TrueType 的技术和设计理念。他在1989年底离开苹果；
- Mike Reed（迈克·里德），主力开发者，并写出了 TrueType 字体设计软件 RoyalT （与 Royalty 同音）；
- Kathryn Weisberg（科特林·魏斯贝格），名义上的项目主管，负责队伍对外的管理事宜；
- Charlton E. Lui（查尔顿·E. 路易），负责 Mac OS 的屏显栅格化；
- Dave Opstad（戴夫·奥普斯塔德），负责排版引擎的行列版式 (Line Layout)；
- Thomas Rickner（托马斯·里克纳），前 Imagen 的 Hinting 技术人员。在 Imagen 的时候，他在著名字体设计师 Charles Bigelow（查尔斯·比奇

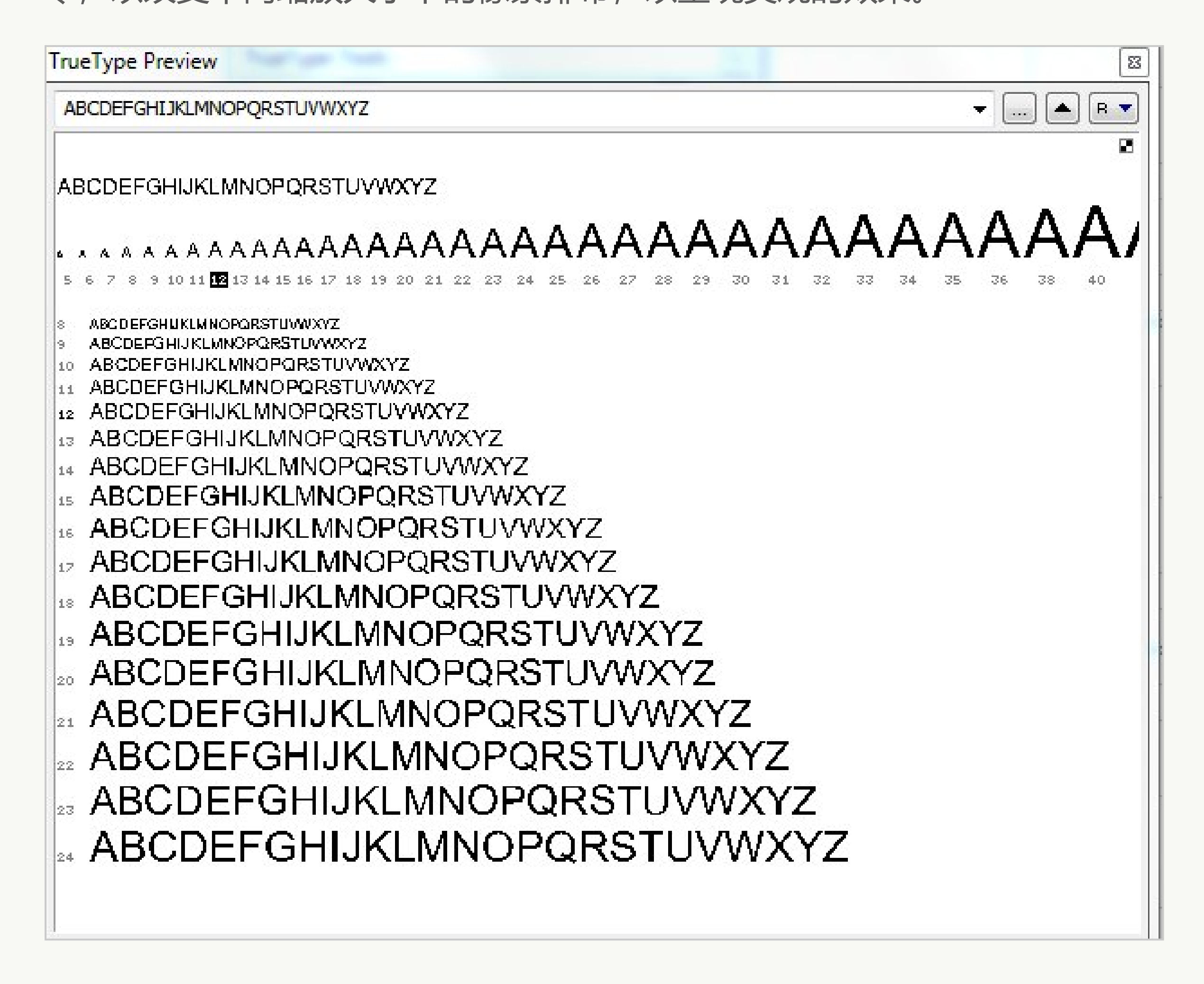
洛）的手下工作。

- David Berlow（大卫·伯洛），外部咨询顾问。他与 Roger Black（罗杰·布拉克）一同成立了 Font Bureau，是拉丁世界首批既精通字体渲染提示技术，又受过传统训练的设计师之一。

4. TrueType 与 PostScript 的异同

与 PostScript 一点不同的是，TrueType 存储字体轮廓数据使用的是二次贝塞尔曲线 (Quadratic bezier curve)。这样的曲线质量与 PostScript 的三次贝塞尔曲线相比，没有那么优秀。然而 TrueType 与 PostScript 最大的不同，在于 hinting（渲染提示）的设计理念。

无论是 TrueType 还是 PostScript 格式的字体，大多都能够做到忠实的轮廓缩放。但是如果仅仅单纯地缩放轮廓（特别是缩小轮廓），在栅格化之后显示出来的图像则会非常丑陋，甚至走形。而渲染提示则是给栅格化引擎下指令，以改变不同缩放大小下的像素排布，以呈现美观的效果。



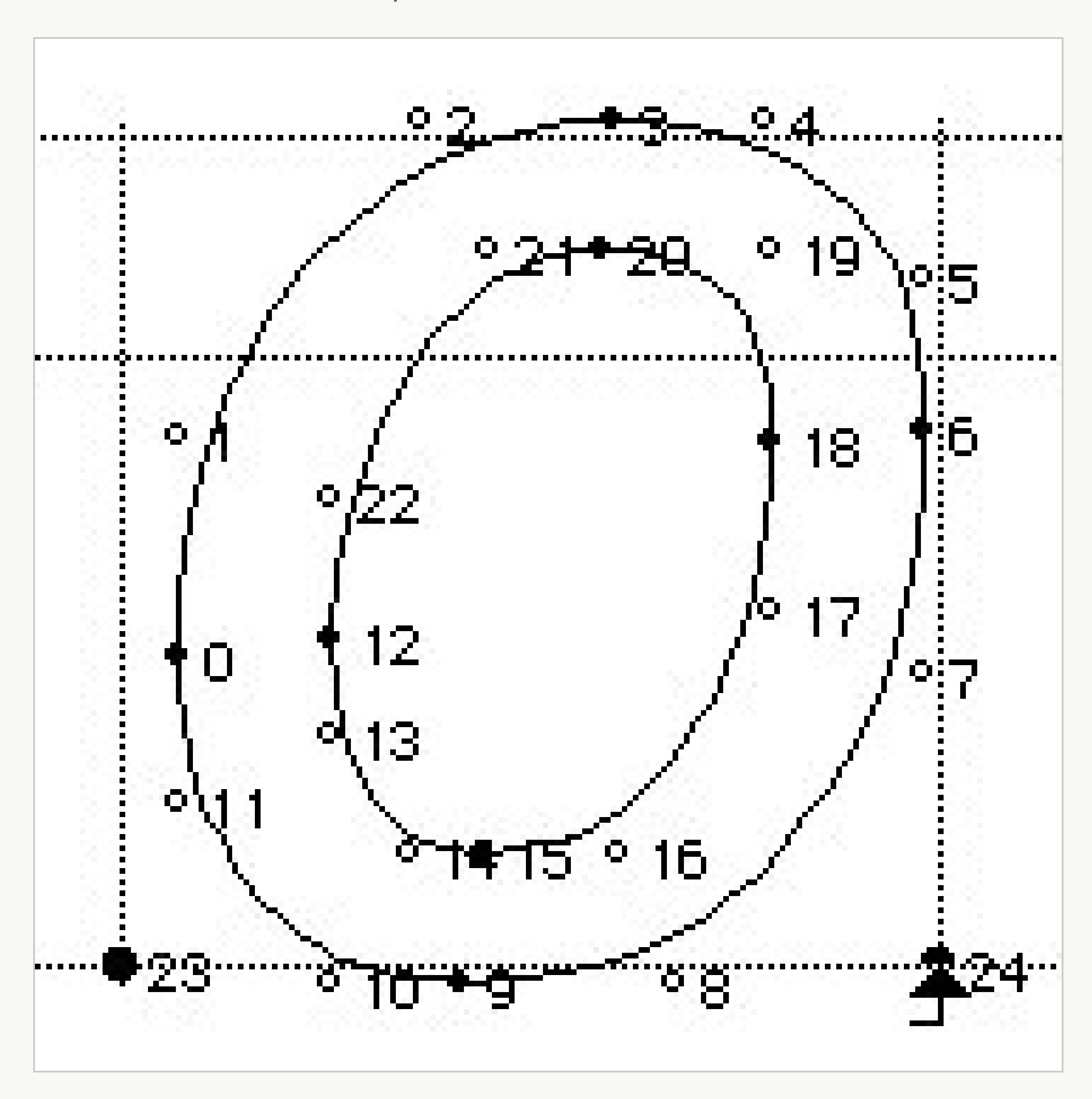
— 未使用渲染提示调整的字，看起来十分丑陋。图：Linotype

由于 PostScript 设计之初就是以印刷输出为优先，因此渲染提示也非常依赖栅格化引擎本身，人称「聪明引擎，笨蛋字体」(Smart rasterizer, dumb fonts)。这也就是说，大部分的渲染提示工作内建在栅格化引擎里面；只要改良软件本身，渲染提示的效果就会得到明显改善，提示的编写也更加概括性。TrueType 则走的是另一个方向，比 PostScript 更加重视屏幕显示。因此，它的 Hinting 不太依赖栅格化引擎，而更依赖人工编写的、精确的、细致的代码。也因此，有些 TrueType 技术人员甚至都不愿意把它叫做 TrueType Hints，而叫做 TrueType Instruction（指令）。自然，TrueType hinting 的理念被称为「笨蛋引擎，聪明字体」(Dumb rasterizer, smart fonts)。

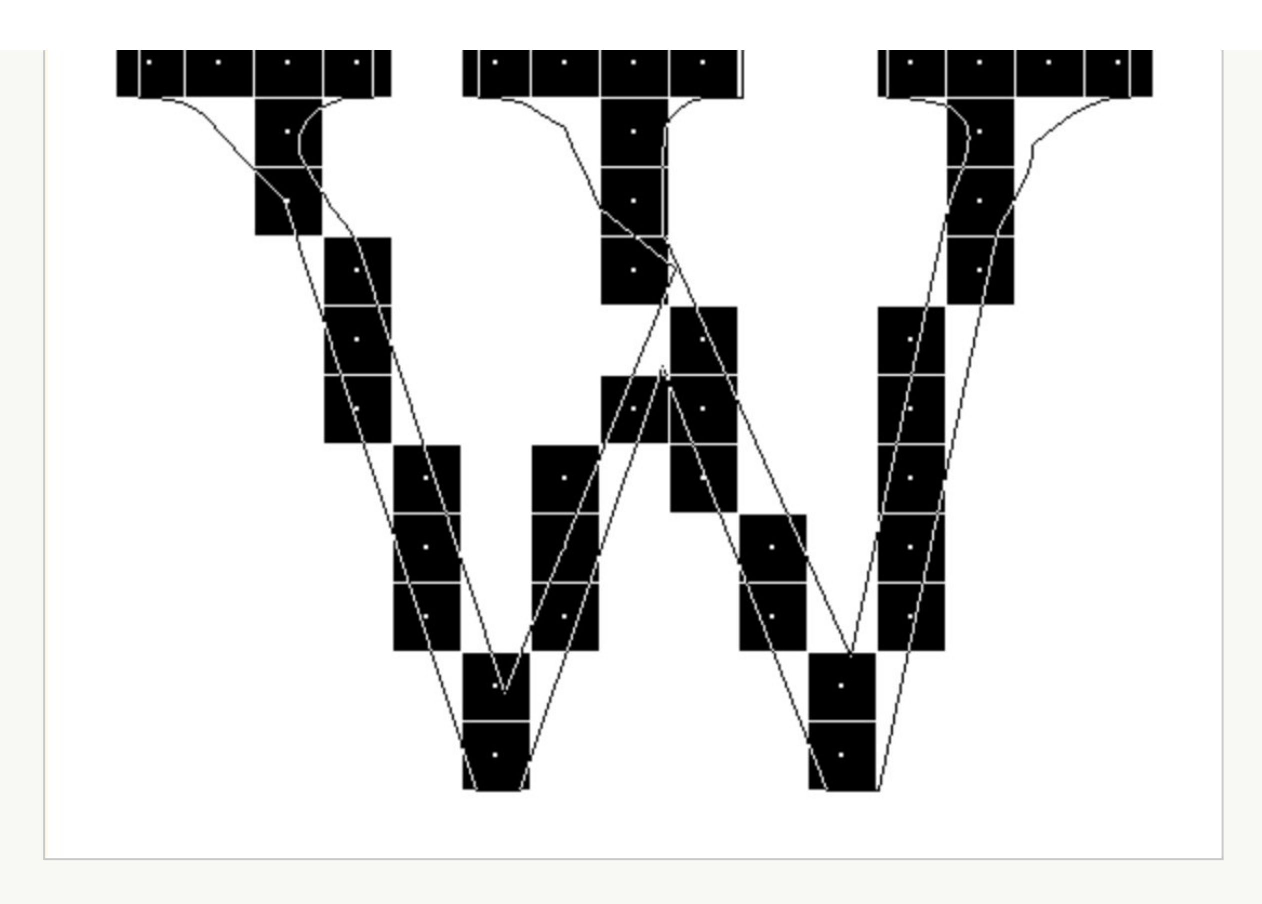
— 带有 Hinting（渲染提示）数据的字。TrueType 和 PostScript 在基本自动 Hinting 上的功能和效果都是差不多的。图：Linotype

TrueType 对于字体中的每一个字形 (glyph) 都有一套与之相对应的「渲染提示」(hinting)。这样看来，一套一套的渲染提示就相当于高德纳 Metafont 中的一个个小程序。主持渲染提示软件设计的 Sampo Kaasila 认为要把渲染提示当成一种编程语言而不仅仅是一种软件参数来做，这样不仅能让渲染提示更灵活、更强大，而且也能够很容易地转换成其他数据存储格式。PostScript 字体的设计师在完成轮廓设计后，在软件设定好各种字体的大概特性，比如升部的长短、x 字高的大小、字干 (stem) 的粗细，然后支持 PostScript 的栅格化引擎会直接按照这些特性，在渲染时自动调整像素的排布。而在 TrueType 中，设计师可以编写一行一行的代码，生成一个 CVT （Control Value Table，控制数值表，也可以随后人手修改），供栅格化引擎调用。

最能体现 TrueType 的人工精细调整策略的，是一种叫做 Delta hint（变量提示）的特殊指令。通常，在完成一套 CVT 之后，设计师便可以检查各个字号下的效果，如果没有丑陋之处，渲染提示工作便完成了。但是如果设计师不满意在某一个字号下当前的像素排布，就可以使用 Delta hint。它能指定一个轮廓上的点，并且把它的位置以邻近两个点的相对位置记录下来，并且可以把这个点向任意方向移动。改变轮廓上的锚点位置，虽然会扭曲字形轮廓，却可以达到开、关某些像素的目的，使得栅格化后的字形在指定字号下看起来美观。这种特殊指令独立于一般的、整体的渲染提示之外，而且仅在某一特定字号下才会被激活。PostScript 则没有这样细致的微调功能。



— 字体的轮廓曲线是由一套点定义的。Delta Hint 可以通过改变这些点的位置，从而扭曲轮廓，达到开关像素的目的。图：Microsoft



— 被 Delta Hint 过的大写 W（来自 Times New Roman，12 Points，96DPI）。可以看到轮廓曲线被明显扭曲，但是所开的像素是好的。图：RasterTragedy

TrueType 的「渲染提示」设计理念，特别是 Delta hint 是一把双刃剑：一方面它给予了设计师更高的自由度、更精细的控制，让 TrueType 格式的字体可以得到更高质量的优化，尤其是屏显优化 [7]；而另一方面，TrueType hinting 这种精确的、细致的调整，需耗费大量的时间和人力物力。尤其是 Delta hint，它只能适用于一个字体文件、一种缩放尺寸。例如，要做从 8 点到 60 点的 Delta hint，设计师要手动调整 52 次。这使得高质量字体的价格开发成本高昂、耗时，价格也非常昂贵。讽刺的是，由于苹果从一开始就开放了全部的 TrueType 技术规范，任何人都可以制作 TrueType 字体。但个人设计师没有时间和精力去做高质量的 Hinting，所以粗制滥造的、盗版的 TrueType 字体大行其道，给 TrueType 这一名字严重地抹了黑 [8]。

读者可能会问，那微软负责开发的 TrueImage 怎么样了——事实是，TrueImage 质量平平，并没有得到苹果（或者其他厂商）的青睐，打印机厂商也不愿意对接一个 PostScript 的仿制品。日后微软私下承认，承诺开发 TrueImage 其实是一种谈判策略，其目的就是为了得到苹果开发的 TrueType [9]。

5. 战争开始

1989 年底，苹果和微软在 Seybold 会议上公布了他们合作开发 TrueType 的商业计划 [10]。TrueType 字体文件格式将会完全公布，不收取任何授权费用；支持解读 TrueType 字体数据的栅格化引擎，会内建在下一代苹果操作系统 System 7（1991 年推出）[11]，也会内建在微软的 Windows 3.1 系统 （1992 年推出）。随后，苹果迅速售出了它持有的全部 Adobe 股份。

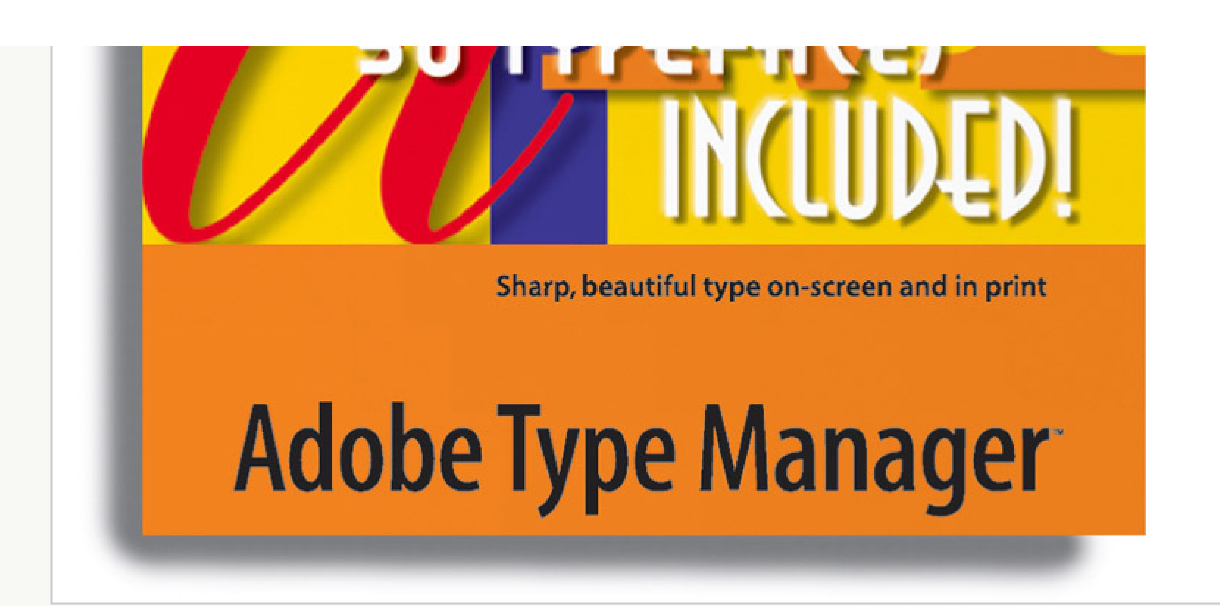
十年字体战争在那年正式拉开序幕。

6. Adobe 反击

1989 年对 Adobe 来说并不是好的一年。一方面，市场上开始大量涌现的仿冒 PostScript 的打印机，使 Adobe 认识到它的垄断地位已经不能维持多久了。而苹果和微软的 TrueType 计划，更迫使它开放了 Type 1 格式的全部技术规范。实际上，那一年 BitStream 字体公司已经成功破解了 Type 1 格式 [12]；即便 Adobe 不主动解密，Type 1 失去独占垄断地位也是迟早的事。

但 Adobe 不会就此甘于失败。Adobe 在 1990 年迅速公布了一款会在将来推出的软件，称为 Adobe Type Manager (ATM)。只要在 Mac OS 上安装 ATM，就能像 TrueType 那样在桌面上即时预览 Type 1 字体。其实在公布 ATM 的时候，这款软件一点都还没写出来；但 Adobe 昼夜赶工，同年就把它做出来了。此时，苹果一方的 TrueType 还没完成。那时还在 Adobe 工作的 David Lemon（戴维·雷蒙） 事后坦言：「我们知道苹果在推出 TrueType 的时候肯定会有这个功能，所以我们提早了九个月把它赶了出来。」





— Adobe Type Manager 软件盒的正面。图：Linotype

但可能 Adobe 最强有力的反击，是由 Sumner Stone（萨姆纳·斯通）带领、在1991年推出的 Multiple Masters 技术。

与高德纳的 Metafont 技术理念一致，Multiple Master（多重母版，下称 MM）是一种利用参数化，允许用户对字形进行插值无级变换 (interpolation) 的技术，是一个 PostScript Type 1 的扩展。例如，极黑 (Ultrablack) 和极细 (Ultralight) 这两个母版组成一对，分别处于一条插值轴 (axis) 的两端。用户只要拥有这两个母版字重，就能自由调配出中间的任何字重，无论是 Light、Regular 还是 Medium。而一个字体中，更可包含不止一条轴。除了字重轴，还有比例轴，可以调配出 Compressed 和 Expanded 的样式；倾斜轴 (slant axis)，可以控制意大利体倾斜的角度；衬线轴，可以让字体从无衬线过渡到衬线体；甚至还有视觉字号轴 (optical size axis)，可以根据字形的缩放，动态地调整轮廓曲线以保证美观和易读性。甚至可以说，只要设计师想，任何可以渐进的变化，都可以用 MM 做到。而 Noordzij 在《笔划》一书中提出的虚拟书写、底下互通的设计理论，也与之遥相呼应。

用来试验、率先发布的字体，便是 Myriad MM 和 Minion MM。随后，Adobe 把一系列旧字体适配到 Multiple Master 格式上，比如 ITC Avant Garde MM、Chaparral MM 和 Adobe Jenson MM。这些字体带有一堆母版，也包含一系列方便用户的预设样式，比如 Regular、Italic 和 Bold。

— Myriad MM；Condensed Light（左下）或者 Semi-extended Black（右上）这些变体，都处在 weight 和 width 这两条轴的相交处。实际上，在 Multiple Master 中，这些角落都需要单独的轴、单独的设计。图：Idsgn

但是，MM 的缺点也非常明显。在 1990 年代，桌上排版这一概念尚未站稳脚跟，从电脑输出到打印的流程也才确立不久。Adobe 既没有大力公关，也没有游说各大软件开发商支持 MM。甚至，因为 Adobe 内部官僚事务繁琐，软件开发团队之间磨擦不断，MM 最终只在 Adobe Illustrator 上获得了全部支持。一个连开发厂商的软件都没有全数支持的技术，是注定没有好结果的。

这也就意味着，要在大部分其他软件上预览和调整 MM 字体，就必须先安装 Adobe Type Manager，然后用它生成一个个独立的字体文件，再安装进操作系统。如果看到不满意，还得把它从操作系统里卸载，再重新生成。这步骤不仅繁琐，而且还制造出一大堆迅速作废的小文件。对桌上排版技术还不大搞得清楚的一般用户到了这里，就已经被吓怕了，也只会用预设好的几个样式。字体设计公司见识到这样情况，所以他们即便使用了 MM 技术来设计字体，也不会把 MM 的源文件直接卖给用户，而是仅仅出售生成好的预设样式。直到 1997 年，市面上的 MM 字体也只有36款，其中过半都是 Adobe 自己制作的。

MM 的另外一个缺陷，在于它的设计理念。由于一条轴是由两个母版定义的，而且只能控制一种轮廓变形，要达到丰富的插值变换，需要的母版数量非常多。实际上，如果有三条轴，就需要有 8 个母版。如果有 n 条轴，就需要有 2^n 个母版。这给设计师、字体技师和用户都带来了麻烦。

7. 苹果的回应

面对 Adobe 来势汹汹的 MM，苹果也感受到了压力。自 1987 年开始、1989 年公布 TrueType 的新闻以来，TrueType 部门就一直在努力工作，以赶上

System 7 在 1991 年的发布。就在发布前的几个月，MM 的消息让 TrueType 部门决定，也要推出支持无级插值变换的功能。他们做到了，这种特殊字体被称为 TrueType GX Variations [13]。

TrueType 部门在了解 MM 的原理后，觉得这样的技术不仅低效，而且要让 TrueType 完全支持这样的技术，也是赶不上发布的。于是，队伍成员开始寻找另一种方法来实现无级插值。他们想到了 TrueType 在设计之初就引以为豪的渲染提示功能。

TrueType Hinting 的本质是通过电脑和人手的控制，在不同字号下改变像素排布，从而达到纠正和提升美感的作用。方法有两种，一种就是通用的、通过栅格化引擎运行的 CVT，直接改变像素的开与关；另一种则是只适用于单个特定尺寸下的 delta hint，它改变的是字形本身的外轮廓，以实现定制的精密调整。

TrueType GX 的主力工程师 Mike Reed 由此提出，既然 Delta hint 可以改变字形的外轮廓，那就可以基于这种概念做无级插值变换。实际上，Delta 一名即意味着「变换」或「微扰」，它记录的是一个点改变的量和方向。而且因为改变的都是矢量，而矢量又可以相互叠加，对字形轮廓的改变是十分灵活的。例如，如果记录下一个字形轮廓中所有的点，从 Regular 到 Bold 的状态变化，又记录下从 Regular 到 Condensed 的状态变化，然后把这两个变化相互叠加，就可以得到 Bold Condensed 的字形。

这种设计理念的聪明之处，在于它完全避开了 MM 需要巨量母版的麻烦。跟 MM 需要把一对母版放在一条轴两端的做法不同，TrueType GX 可以把起始的母版（可以是 Regular 也可以是 Medium）放在轴上的任何一个地方。极值的母版（比如 Ultrablack 和 Ultralight）则完全可以通过与起始点的变化差别来表示。这样就可以做到多条轴共用一个起始点。实际上，在 TrueType GX 中，如果有 n 条轴，只需要 (n+1) 个母版，这比起 MM 的 2^n 个，实在是简洁太多了。更厉害的是，TrueType GX 的特性可以把多条轴的参数变化叠加。也就是说，只要有了字重轴和比例轴，就可以直接得到 Bold Condensed 或者 Light Expanded 这样的多重插值（在 MM 上，这需要另加一条轴）。

另外一个 TrueType GX 的优势，是可以把 Hinting 数据也对应到轴变换上。既然 CVT 中记录的数值可以指导栅格化引擎如何排布像素，那么这种数值自然也是可以根据尺寸而变换的。

在 TrueType GX 可变字体中，存在以下这几种关键的表：

- glyf：从普通 TrueType 字体继承。存储字形；
- fvar：Font Variation 的简称。存储轴的信息；
- gvar：Glyph Variation 的简称。存储字形在轴上的变化信息；
- cvar：CVT Variation 的简称。它让 CVT 里的数值也可以跟随变换；
- avar：Axis Variation 的简称。它可以自定义非线性的轴变换。简而言之，可以改变在轴上变换时，字形轮廓变化的各部分速度；
- fmtx：Font Metrics 的简称。它对应的其实是 gvar 表中的数值：当 gvar 指示的字形变化达到制定数值时，fmtx 表可以更改字体的 Metrics。这可以让不同的视觉字号 (Optical Size) 使用不一样的 Metrics。

最早支持 TrueType GX 可变技术的，大致有两款字体：Jonathan Hoefler（乔纳森·赫夫勒）受苹果邀请而设计的 Hoefler Text，以及苹果与 Matthew Carter（马修·卡特）合作设计的 Skia。它们直到 macOS Sierra 仍然是系统的预装字体。

除此之外，苹果开发的 GX 排版引擎也拥有一系列高级排版特性。一个最有力的例子是，在 Adobe 的排版软件里，用户需要删掉 fi 的文本并且手动插入 f-i 合字这个特殊字符，也就是改变了语义。与之相对，GX 配合 TrueType 字体格式，率先支持了行内热替换。也就是说，排版引擎可以在不改变源文本的情况下实现 f-i、f-f-i 这些合字，并且还能不影响拼写检查，更可以提供多级别热替换的细微控制，比如 Contextual Alternates（上下文变体）。这在今天的 Adobe InDesign 甚至是 Microsoft Word 都能实现，然而在当时则远远领先。

8. 微软倒戈

苹果 GX，无论是排版引擎还是 TrueType 可变字体格式，在那个年代都是相当于黑科技一样的存在：编程上高效、功能上先进、设计理念在字体和排版设计的角度看来，更是高瞻远瞩。

那时使用桌上排版的用户，在那时大多都是没有经过专业训练、没有深厚背景知识的普通人，不仅无法理解这些高级功能的精妙，而且只关心的是容不容易理解、快不快上手。可能也是因为如此，苹果决定，这样的先进科技应该直接内建到系统本身。任何的热替换和高级塑形功能 (Advanced Shaping) 都需要自动进行，不需要用户操心。所以它也没有向普通用户大力推广这些功能的好处，也没有开发好用的字体设计软件。

然而苹果的善意，恰恰导致了 GX 和 TrueType 可变字体的悲惨命运。因为没有好用的软件，字体设计师没有办法迅速利用 TrueType GX Variation 的优秀特性，于是为 TrueType GX 设计的可变字体少之又少。在排版软件方面，出现了这样一种讽刺现象：设计软件厂商之前花费了大量精力去独立开发高级排版功能，比如（自动齐头尾，Automatic Justification），而苹果把原来可以收取昂贵费用的高级功能直接集成到系统里，让用户随便使用，这是变相砸了他们的饭碗。因此，即便苹果 GX 比排版软件自带的还要优秀，大部分设计软件仍然不愿意接入系统自带的文字处理功能。至于 Adobe，更是借机报复。它收购了 PageMaker 排版软件的公司 Aldus，于是 PageMaker（以及以后的 InDesign）一直都不支持使用 GX 套件来排版。一个既没有可用字体，设计软件又不支持的文字处理技术，无论有多优秀，终究会被埋没。况且，PostScript 因为起步时间早，在打印机厂商中也获得了充分的支持。即便字体和排版设计的软件支持了 GX，打印机也需要另外更新支持 GX 的驱动，这对打印机厂商来说成本和风险都太高了。

这时，苹果眼看着微软通过售卖捆绑 TrueType 技术的软件（大多是操作系统）获利，也想从中分一杯羹。因此，在微软要求得到 TrueType GX 授权时，苹果拒绝了。这可能是因为它想要像微软一样，通过捆绑 TrueType 来吸引用户付费升级系统。但这一步却加速了 GX 和 TrueType 的衰落：软件厂商是不会支持一个不兼容 Windows 的标准的。

被拒绝的微软，马上转向了 Adobe 的怀抱，联合起来对付苹果。这一点不无讽刺：几年前，苹果正是被 Adobe 拒绝，才与微软结盟。Adobe 与微软结盟后在 1997 年发布了 OpenType。第一款真正的 OpenType 字体由微软与 Hermann Zapf 合作：Palatino Linotype，随 Windows 2000 预装。

— Palatino Linotype。微软既想让 OpenType 的技术得到充分的炫耀，又想搞好与 Hermann Zapf 的关系，便从 Linotype 处得到授权，在 Zapf 的监督下，扩充和重设计了 Palatino。这一版本的 Palatino 使用了包括各种热替换在内的许多 OpenType 特性，同时也支持大范围的重音和变音符号。图：Microsoft

9. 字体战争的尾声

微软与 Adobe 的结盟，除了苹果的拒绝，还有其他因素。因为在 Windows 上支持了 TrueType，而 TrueType 的名声被抹黑了，所以微软在设计师和图形艺术家之间的声誉一直不太好；支持 PostScript 字体格式，也就攀上了 Adobe 的名声，可以有效改善这一问题。Adobe 则意识到，Type 1 格式本身的技术相对落后，导致卖字体越来越难；与微软合作，则可以打开销路。于是混血儿 OpenType 诞生了：它的基础来自 TrueType，继承了所有 TrueType 的先进理念和软件设计思想。同时，它又是一个容器 (wrapper)，可以贮存 TrueType 和 PostScript Type 2 格式的轮廓曲线。并且，它兼容 Unicode，意味着它最多可以支持 65535 个码位。这对一心想要支持各种非拉丁复杂语系的微软，是非常有吸引力的。

在与微软达成协议后，Adobe 痛定思痛，准备全力推广 OpenType 标准。一心不能二用，要让文字部门的全体人力都投入到新标准的推广中，就必须砍掉 MM。于是 Adobe 花了两年时间，把 MM 字体转换成 OpenType，项目代号是 Alchemy [14]。Adobe 派人到各大会议、各大字体设计公司，不遗余力地进行宣传。这样的公关战略打得非常成功，市场对 OpenType 的接受度也越来越高。

虽然 MM 作为一种终端用户格式被抛弃了，它的技术依然在各大字体设计软件中保留着。大多数字体设计软件如 FontLab 和 RoboFont，都支持编辑 MM 字体格式，然后输出生成好的结果 (instance)，这也是多字重、多样式的大字族设计的一般流程。甚至还有专门的软件如 Superpolator，专门负责插值生成字体。

苹果在 Mac OS 8 就已经移除了 GX，并在 10.4 加入了对 OpenType 的系统支持，象征着它在这场字体战争中的失败。但即便如此，TrueType GX 的遗产，特别是 TrueType GX Variation 可变字体格式，都在苹果内部继续流传。TrueType GX 的编码支持部分被改名为 ATSUI（Apple Type Services for Unicode Imaging，苹果万国码文字成像服务），另一部分被改名为 AAT（Apple Advanced Typography，苹果高级文字排版技术）。从 iOS 7 就新增的 Dynamic Type，可以实时调整整个 iOS 系统的字体大小，也支持视觉字号的变换，就是运用了 AAT 技术。AAT 的高级塑形、高级定位功能 (Advanced Shaping & Advanced Positioning) 后来更发展出对复杂书写系统的支持，特别是印地语系。

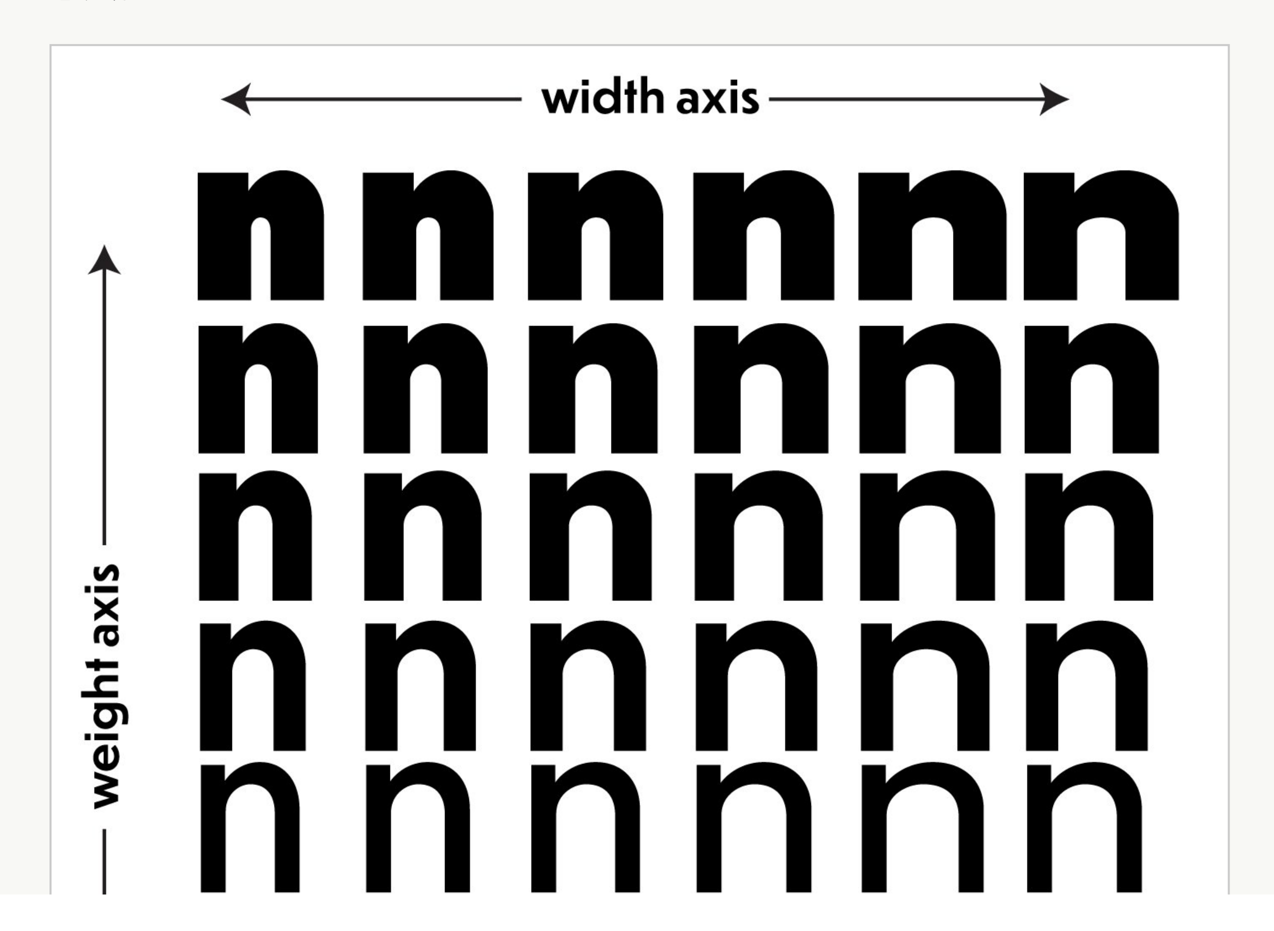
从 1987 年 TrueType 开始开发，到 1997 年 OpenType 发布，这场字体战争打了十年才终于落幕。期间 MM 和 TrueType GX Variation 这两种无级变换技术，成为了这场战争里的技术明星。MM 理念先行，然而在设计和技术上都有缺陷，使用麻烦，最终被抛弃；而 TrueType GX Variation 比起 MM，设计、理念和技术实现上都优秀许多，但因为苹果的封闭作风和不当的商业时机，也没得到广泛应用。

时过境迁，18 年过去了。在这期间，个人电脑迅速平民化，手机和网络的发展也异常迅猛，在网页上实现高质量排版设计的呼声也日益增长。不少新一代设计师在了解到 MM 和 GX 之后都会感叹，要是这样高瞻远瞩的技术也能运用到当今的 OpenType 上就好了。

参数化设计和无级变换的梦想，在 18 年前远去。在 2016 年的华沙，它回来了。

10. OpenType Variable Fonts

近年来，在网页上实现高质量排版设计、让设计师拥有更高的自由的呼声，随着移动设备的广泛普及和 CSS3 webfont 的使用，也越来越高。不少人在阅读这段设计历史后，都萌生了想复活 MM 或者 TrueType GX Variation 的想法。



nnnnnn
nnnnnn

— 字体在字重轴、比例轴上的变化。图：CJ Dunn，字体 Dunbar

但其实在 2013 年左右，以 Peter Constable 为首的微软文字团队，便已开始研究无级插值变换的想法。他们的动机就是两个：给予用户更高的自由度、更精细的控制；减少各类字体文件占用的体积，无论是网页加载，还是小内存设备的字体内嵌。在 2015 年，微软的团队与苹果进行了洽谈，希望把 TrueType GX Variation （现 AAT）整合到 OpenType 中，并获得了支持。2016 年 1 月，一位并不知情的谷歌开发者 Behdad Esfahbod （贝赫达德·埃斯法赫博德）发表了 OpenType GX 的提案，而微软和苹果的团队随后联系上了他。2 月，在获得了 Adobe 的支持后，微软、苹果、谷歌、Adobe 的四方联合工作组初现雏形。从 3 月到 7 月，工作组多次会面，确定了大致需求和发布的时间表。

2016 年 9 月在华沙召开的第 60 届 ATypI 年会上，OpenType 1.8 正式发布。它基于的不是 Adobe 的 MM，而是苹果贡献的 TrueType GX 技术。谷歌开发了一个可以在 Windows 上实时预览 1.8 版字体的小工具叫 FreeView。微软在 Windows 10 的周年庆典版 (Anniversary Edition) 的 Edge 浏览器中，已经率先支持了新版的规范。而苹果则走得更远：macOS Sierra 中的系统字体 San Francisco UI，已经变成了 OpenType 可变字体。在会议召开的同时，相关人士也着手与 W3C 的 CSS 工作组洽谈，以求让新规范更快获得广泛的浏览器支持。

ah! ah! ah!

xheight axis

— 字体在 x-height 轴上的变化。图：CJ Dunn，字体 Dunbar

由于继承自 TrueType GX，所以 OpenType Variation 的大部分特性，都与 18 年前的 GX 一脉相承。这些特性主要的有：

- 从单一套轮廓设计开始。这个起始的轮廓包 (Outline Set) 并不一定非要是 Regular，也可以是 Semilight 或者 Medium。新增的极限设计 (Extreme) 或者任意一个中间设计，都可以通过与起始轮廓的变化量来表示。因为主要占体积的只有起始轮廓，这种只储存变化量的策略还可以极大压缩多字重、多样式字体家族的整体体积。
- 这些轮廓点上的变化量以变量包 (delta set) 的方式贮存。也就是说，每一个不是起始的设计，都是一个 delta set。需要注意的是，delta set 与 delta hint 是不一样的，但 delta set 的概念，是从 TrueType delta hint 发展而来。（实际上，开发者们强烈不推荐在 OpenType Variable Font 里做 delta hint，因为「某个特定字号」这样的概念不再存在，所有改动都是全局的）
- 每一个新增的极限都可以与起始轮廓形成一条轴。与 TrueType GX 一样，要有 n 条轴，只需要 (n+1) 个设计，而不是像 MM 那样需要 2^n 个设计。
- 忠于 1989 年 Sampo Kaasila 和 Mike Reed 的理念，使得矢量变化可以相互叠加。插值生成的角落（比如字重轴和比例轴相交的角落）不需要新增设计，而可以让软件自动生成。当然，如果设计师对此不满意，也可以在角落新增一个设计，施加更细致的控制。
- 几乎完全继承 TrueType GX 的数据表，比如 fvar、gvar 和 cvar。有了 cvar 表，就意味着插值变换的过程可以不是线性的，设计师可以细致控制插值变换的速度和量。fmtx 表被换成了 mvar (Metrics Variation) 表，与原先设想的功能一致。
- 新增一个 rvrn 表（Required Variation Alternates）。此表是为了大量非拉丁语言所准备，它可以在文字处理引擎开始工作之前就选择好相应地区对应的字形和字符 [15]。
- 在普通的 TrueType 字体中，Hinting 需要重复做，这一点已经在前文解释过。OpenType Variation Fonts 由于有了 cvar 表，可以把 Hinting 数据对应到轴变化上。这也就意味着，Hinting 只需要做一次，就可以伴随着各种轴变化和字形缩放而变化。
- 新增 stat 表，以支持 Variable Font 下的字体名称显示。此表可以记录一

些预设的设计（比如 Light 和 Medium 这种用户常用，但又不在起始轮廓或者极限上的设计），不仅可以让不支持 OpenType Variations 的软件正常读取，也为字体命名带来全新的理念。

- 支持自定条件下的字形替换 (Conditional Glyph Substitution)。例如，当字重改变到一定程度时，字形本身的结构或骨骼也需要改变，比如大写 Q 的脚需要不穿透圆形内部。如果使用轮廓变化来做，那么在轴上的某些点可能就会出现奇怪的字形。为了解决这个问题，Variable Font 支持设计好另外一个可供替换的字形，并且默认隐藏，只有在达到插值变换到一定程度时才会激活，直接替换原有的字形。不仅可以更美观，同时这种功能也可能会对非拉丁的复杂书写系统设计作出可观的贡献。

中文读者也可以想象一下这样的可变字体对东亚文字设计的影响，比如：

- 与西文混排时的灰度调整、比例调整，将会变得非常容易。在现在，网页设计师为了与拉丁字体相配，被迫采用寥寥几个多字重东亚家族，比如思源黑体。如果使用支持 OpenType Variation 的东亚字体，便可以在页面上自由地调整字重和笔划的粗细。
- 在网页上，只需要加载一个字体文件，通过 CSS 的控制，就可以达到同一网页、多字重、多样式，这在之前是完全不可能的。甚至为了进一步压缩文件体积，还可以只使用各种轴的半边（比如只使用 Regular 到 Bold，而放弃向下到 Light 的那半条轴）。
- 中西混排的垂直对齐调整，一直是只有 InDesign 类专业软件才具备的功能。因为 OpenType Variable Font 可以很容易地把字体的垂直属性（比如 x 字高和升降部的比例）设成轴，在未来可能就会看到浏览器和普通文档编辑器做到这些细微的调整。
- 对东亚字体做细致的、全家族覆盖的 Hinting，在以往需要耗费大量的人力物力，所以也只有微软雅黑这样的字体能负担得起。有了 OpenType Variable Font，只需要做一次渲染提示，而且可以在几个设计师的能力之内完成。再配合例如 ideohint 之类的自动化工具，一个设计师完全完成也不是不可能 [16]。
- 可能是因为开发成本太高，东亚字体的各种 Optical Size（比如书宋和标宋）一直以来都没有成系统。OpenType Variable Font 可以让字体设计师以一个起始轮廓设计为基础，容易地设计出全套 Optical Size 的覆盖，并且还能做到内建拉丁字形的变化、 Font Metric 和 Hinting 的跟随。甚至设计师还可以直接使用视觉字号轴，在小字号下改变字形的轮廓曲线来提升易认性（legibility），而不需要像以前那样借助于 Hinting。

- 与拉丁字体设计相比，东亚字体设计的风格化也进步得非常缓慢。这也是由于超大字符集带来的开发成本太高、市场风险太大所导致。OpenType Variable Font 的出现，可以有效降低开发成本，让市场探索和字体设计产业的前进更迅速。在试验、探索和改进的成本降低后，东亚字体厂商可以从保险的黑体和宋体这些传统流派里解放出来，推动东亚视觉设计的多元化风格发展。同时，它也让稍作改动就拿出来当新字体售卖的盗版设计无所遁形。
- 在混有拉丁字母和标点符号的中文版面上，要遵循传统的齐头尾美学标准，一直是一个大难题。在东亚字形尺寸固定的情况下，要么齐头尾会有时候变得非常难看，要么设计师就被迫使用齐头散尾。OpenType Variable Font 可以极细微地压缩和拉伸字形的横向比例，并且不致丑陋。这样，在视觉上没有明显变动的情况下，设计师可以更灵活、更精细地调整一行字，从而在混排中也能更容易达到美观的齐头尾。
- 如果一款字体需要同时适配纵向和横向排版，它的比例轴可以与其他 OpenType 特性相结合，从而实现在纵排时变扁、在横排时变高等等特殊优化效果。
- ……

当然，有了用户端、运行时的插值变化，并不能够完全解决所有的问题。甚至，新技术还会带来一系列新的问题，最明显的就是如何给这种字体定价。而 Variable Font 对于打击盗版、推广正版字体的影响，到现在也还不能说清。诚然，这一次 OpenType Variation 的制定和发布，是四家大公司共同参与、获得广泛支持的。但长远来看，也无法推测在往后的几年到十几年间，它们在自己的软硬件产品上对 OpenType 1.8 的支持和实现会是如何。最重要的是，可变字体能不能获得一般用户的热烈欢迎，四大公司在研发和推广上投入的人力物力能否经受得住投资者的审视，这关乎 OpenType Variation 的生死存亡。

从高德纳和 Gerrit Noordzij 铺平道路开始，参数化字体设计和用户端的无级插值变换，一直都是字体业界的追求。无论是带有繁复合字的希腊文活字，还是连绵飘忽的Kazuraki，都体现一个做字人几百年来藏在心底的梦：字不再局限于一个个小铅块、一张张小胶片、一个个小方格之中，而是灵活地、流动地、精细地随意变形 [17] [18]。

上个世纪的普通用户未能理解到无级插值变换的精妙，具体实现上的缺陷和商业公司的明争暗斗，也把它暂时扼杀。而在民众的排版设计意识逐渐浓厚、字体技术也有了长足发展的今天，可变字体的梦想，终又浴火重生。ATypI 华沙的发表会，是向这梦想进发的第一步。

征途刚刚开始。OpenType Variable Font 在未来的影响和命运，让我们拭目以待。

注：

1. 实际上，早在 1970 年，德国 URW++ 的 Peter Karow 便已使用了无级插值 (Interpolation) 技术，并把它应用到 URW++ 独家的软件 Ikarus 中。但 Ikarus 仅仅是用于把活字数码化，其软件设计的理念并没有 Metafont 那样超前。
2. Knuth 本人也承认: Asking an artist to become enough of a mathematician to understand how to write a font with 60 parameters is too much.（要让一个艺术家成为一个数学家，然后编写一个带有60个参数的字体，太难了。）
3. 后来高德纳的朋友改造了 Metafont，使之能够直接输出 PS Type 1 字体文件，这套改造过的 Metafont 被称为 MetaPost。
4. CFF 全名为 Compact Font Format（紧凑型字体格式），是 Adobe 为了解决大字符集的性能问题而开发的。
5. Type 42 是一种转换格式，支持把 TrueType 轮廓的字体转换到支持 PostScript 的打印机上打印，这应该是 Adobe 为了抢夺 TrueType 的市场而开发出来的。不同厂商格式相互转换之难，可能也是引用《银河系漫游指南》的原因。此外，Type 42 在应对东亚文字（中日韩 / CJK）的时候性能也有大幅提升。
6. Aldus 十几年间日渐式微，对 PageMaker 的开发也逐渐停止。Adobe 后来将 Aldus 并购，在 2001 年左右停止了对 PageMaker 的支持，并把大多数 PageMaker 的功能转移到 Adobe InDesign 2。
7. 有人甚至认为，TrueType 字体的价格不是来源于轮廓曲线的设计，而是来源于 Hinting 数据。
8. 另一个抹黑的因素自然来源于它所使用的二次曲线，其质量不如 PostScript 的三次曲线。然而这并不是主要原因。现在想来不无反讽：为了赚钱的 Adobe 把 Type 1 的技术规范保密，反而遏制了丑陋业余字体的泛滥。
9. 这是笔者在 2016 年的 TypeCon 大会上听微软文字部门的成员 Greg Hitchcock 说的。
10. 一个桌上出版的设计和技术会议，于 2000 年左右停办。
11. Kaasila 设计的 TrueType 是 32 位的，然而由于技术当时还不够完善，Windows 3.1 内建的 TrueType 默认使用的是 16 位的栅格化引擎，一直到 Windows NT 3.1 才换成默认 32 位。
12. 由著名字体设计师 Matthew Carter（马修·卡特）创立。
13. 开发 TrueType GX Variation 的项目内部代号是 Bass（鲈鱼），因为鲈鱼是「可以刮鳞的」（Scale-able），而 Scale 又有「缩放」的定义，也就是「可以缩放的」。

14. OpenType 早期曾经想过加入类似 MM 的无级变换功能，但可能是因为前车之鉴还历历在目，因此很快又移除了。
15. 这新增的表可能就是 John Hudson 的主意。他一直热心关注在非拉丁复杂书写系统中，排版引擎需要作什么样的变化来适应。他的理论发表在这里。
16. 由中国技术员 Belleve Invis 开发，原名为 sfdhanautohint。
17. 此处展示的是 16 世纪刻字工 Robert Granjon（罗贝尔·格朗荣）的希腊文活字。为了最大程度地还原希腊文书法中飘逸的连笔，Granjon 刻制了数量惊人的合字。
18. 由西塚凉子设计。Kazuraki 原名 Teika（即「定家」），是基于藤原定家诗集中的行书字形创作。它是 Adobe 炫耀 OpenType 技术的字体，并于 2010 年获得 TDC2 大奖。

参考文献

1. MetaFont, *Cunningham & Cunningham, Inc.*
2. John Hudson, Introducing OpenType Variable Fonts, *Medium*
3. Nick Sherman, Variable Fonts for Responsive Design, *A List Apart*
4. Roel Nielskens, Variable Fonts: the Future of (Web) Type, *Typographica*
5. Tamye Riggs, How the Originals Endured an Ever-Changing Industry, *TypeKit*
6. Laurence Penney, A History of TrueType, *TrueType Typography*
7. Thomas Phinney, TrueType & PostScript Type 1: What's the Difference? *TrueType Typography*
8. A Talk with Sampo Kaasila, *TrueType Typography*
9. 现代版 Ikarus，由 Dutch Type Library 与 URW++ 共同继续研发, *Vimeo*
10. David Berlow, OpenType Font Variations Open Up a New World of Possibilities, *Type Network*
11. Beat Stamm, The Raster Tragedy at Low-Resolution Revisited: Opportunities and Challenges beyond "Delta-Hinting"
12. Palatino Linotype, *Microsoft Typography*
13. Timothy Ahrens, A Closer Look at TrueType Hinting, *TypeKit*
14. Functions and Delta Hints, *Microsoft Typography*
15. Peter Bil'ak, Font Hinting, *Typotheque*
16. Apple TrueType Reference Manual, *Apple*
17. ATSUI, *Apple*
18. Typo Berlin 2014: Behdad Esfahbod, *Type Talks*
19. Typo Berlin 2014: Thomas Rickner, *Type Talks*
20. ATypI 2017，OpenType Variation 发布会视频, *YouTube*
21. OpenType Font Variations Overview, *Microsoft*
22. 暂定版 OpenType 1.8 规范, *Microsoft*
23. CSS 工作组提出的、支持 OpenType Variable Fonts 标准的草案, *CSSWG*

24. [OTVar] Introducing OpenType variable fonts, TypeDrawers
25. FitFont, Type Project
26. 本文作者特别感谢 John Hudson 先生提供技术细节的解答。

尊重原创：关于转载

我们希望在中文环境中建立一种健康的 TrackBack 和链接机制，保证原创，并不影响传播。因此对于译文和原创文章，我们欢迎您在网站上推荐我们的文章，包括文字和图片片段，但不赞成任何形式的全文转载。

参与讨论或通过 TrackBack 推荐：Trackback URL.

12 个相关讨论

Belleve
2016/09/28 at 10:00 pm | Permalink

那啥，ideohint 做不了 VF 的 hint 的，因为极小尺寸的位置决议和宽度决议需要大量的 AI 算法计算，而且牵涉到笔画合并，不好做的。
不过能支持 VF 的环境里面基本上也没必要做 hint 了吧

Belleve
2016/09/28 at 10:03 pm | Permalink

另外 VF 对于变拓扑和一些高级建模（如基于笔画的建模）支持仍然有限

張玉貞
2016/09/28 at 11:16 pm | Permalink

這篇文章有頭有尾真是太精彩了，讓繼續喜歡業餘網頁設計師的我看得津津有味，你下的功夫真是非常龐大，致敬並且致謝。因為利用Wix套版做網頁，他並沒有專屬的中文字體、非常痛苦只依附於西方字體、變化少的可憐。如果這次四大合作能有更自由的變化，那未來版面設計就不一定是西方思考的天下、也不必苦於英文字比較美、但是中文夠美的字體不多的痛苦

Jiang Jiang
2016/09/29 at 12:05 am | Permalink

"后来高德纳的朋友改造了 Metafont，使之能够直接输出 PS Type 1 字体文件，这套改造过的 Metafont 被称为 MetaPost。"这个说法不对，参考 https://www.tug.org/TUGboat/tb19-3/hoek1.pdf 和 https://www.tug.org/TUGboat/tb24-3/jackowski.pdf MetaPost 只是辅助生成 stroke 而不能直接生成字体文件，MetaPost 的创制也不是为了制作字体而是为了绘制图表。

孫志貴
2016/09/30 at 5:27 pm | Permalink

接下来就看 Apple 有没有意识将 macOS / iOS 的中日韩介面字型改用同一个可变字体家族了。

Chaojie Mo
2016/10/07 at 1:28 am | Permalink

"在推出 LaserWriter 的同一年，苹果与当时的排版软件公司 Aldus 会谈（Aldus 之名，即来自文艺复兴时期意大利书法家阿尔杜斯·马努提乌斯 Aldus Manutius）。"
此处将 Aldus Manutius 称为书法家实在是非常不准确的说法，Aldus Manutius 只是印刷商和出版商，Adline 出版社的字体主要是 Francesco Griffo 的作品，而连 Griffo 都不能说是书法家。也许作者是将同时代的书法家兼字体设计师 Arrighi 和 Aldus 记混了？

史铭
2016/10/19 at 5:02 am | Permalink

看到楼上的有些评论只专注一些技术细节或一些历史出处的讨论上，太吹毛求疵了，完全没有意识到作者站在整个宏观角度讨论这件事的意义。一件重大历史意义的大事，就这么悄无声息的展开了，可能今天的我们仍无法估量它给我们带来改变的意义，TrueType GX Variation不只是一个单纯技术上的整合，一切都是屏幕尺寸变化适应，人们新阅读习惯改变，而给出的全新的解决方案，

这些挑战，在传统纸质书上是不会遇到的。所以，真的非常精彩的文章，花了几个小时一口气读完，为作者专业的态度而致敬，脉络清晰，上下文的历史背景介绍的非常到位，特别是最后对中西文混排的预见的分析很受启发。

谭宏涛

2016/11/17 at 10:38 pm | Permalink

非常精彩的文章，好多猛料。一口气看完了。不知道是翻译的还是原著呢?

虎卅

2016/11/18 at 1:33 am | Permalink

感謝作者講述背後的風雨、看完覺得莫名的感動呢!

Krasnaya Ploshchad’

2017/03/01 at 6:19 am | Permalink

当初 OpenType 问世以后 Adobe 不光派人做宣传，后来还向 ISO 组织申请了认证，代码是 ISO/IEC 14496-22。

Krasnaya Ploshchad’

2017/03/27 at 11:24 am | Permalink

中西混排的垂直对齐调整其实是可以不用 OpenType Variation 就能搞定的，只要软件支持 OpenType 的 BASE 表就能根据这个表的数据重排。

Krasnaya Ploshchad’

2017/12/27 at 2:36 am | Permalink

微软在 Windows 10 秋季创意更新版里面就添加了一款可变字体 Bahnschrift，有 5 个字重可用。

4 个Trackbacks

By Google 的新字体可让你随便改，而且还能拿去卖 – 站点标题 on 2017/06/24 at 8:08 am

[…] 相对而言，手迹造字更合适字体爱好者，如果设计师想用这个来细细调节字体就不行了。据了解，目前为止还没有参数化中文字体可供进行自主调整使用。一直以来，东亚字体的发展都比较缓慢，毕竟和拉丁字母相比，东亚字体的体量大很多，开发成本和风险也更高。 […]

By Type is Beautiful » TypeTime! 文鼎字体设计沙龙及多媒体展览 on 2017/10/10 at 6:36 pm

[…] 年，OpenType 1.8 规范宣布了支持可变字体（variable font）功能。而文鼎字库此前在晶熙黑字体上运用推出的 Infinity Font […]

By Type is Beautiful » 字谈字畅 085：字体之秋·二〇一八北京篇 on 2018/10/30 at 10:24 am

[…] TrueType 简史；谭沛然所撰〈参数化设计与字体战争：从 OpenType 1.8 说起〉一文中，亦有字体格式发展历史的相关介绍 […]

By 一派·Podcast｜關於「可變字體」的閑聊與科普 – 唐人街新聞 on 2021/04/16 at 5:35 pm

[…] 《參數化設計與字體戰爭：從 OpenType 1.8 說起》 […]

参与讨论

你的Email地址将不会被发布或透漏。 标记*的项目为必填项目。

名字 *

Email *

网站

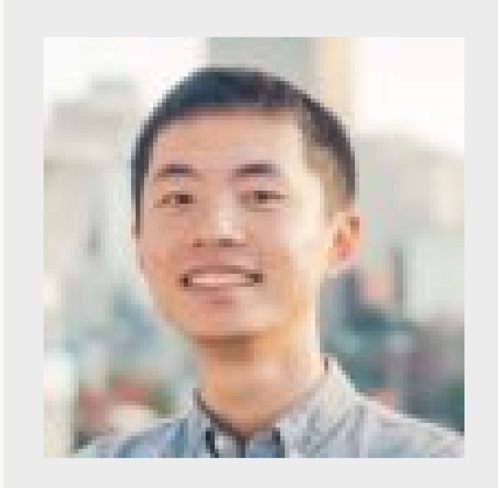
评论

发表

« Eric Liu 东京讲座：中西混排和日西混排的思考 3

2016 TypeTour Tokyo 总结 »

作者 / 译者



谭 沛然

The Type 编辑，交互和平面设计师、设计史研究者。

Twitter

最近文章

Vojtěch Preissig：一名爱国者

十年磨砺之作：复刻字体 Heldane

透明、无菌、纯粹：现代主义设计的秩序迷恋

末世感叩击：《新世纪福音战士》的文字世界

Steglitzer Werkstatt：现代商业设计事务所的初露头角

本站

文字设计和视觉文化
我们关注的话题包括字体、排版、平面设计、公共设计、技术和视觉文化。关于 »

相关书单

基础文章汇总

Twitter / Facebook / 微博 / 微信

会员计划：
即刻加入

The Type Members

播客：
字谈字畅



孔雀计划：
中文字体排印



包豪斯以外：
现代主义的另一面

上海活字
社会、历史与人文

活字字体的基础讲座
译自 欣喜堂

关联书籍

《西文字体的故事》

《字体故事》（*Just My Type*）

《西文字体》

《西文字体 2》

相关文章

說 Helvetica 和 Arial

Myriad 的故事

WOFF 使用指南

字体数字化简史与 WOFF

Monotype Web Fonts 测试

Computer Modern 的故事

FontExplorer X for PC Beta1 版提供下载

思源黑体（Source Han Sans）发布

关于车辆号码牌字体

Justfont：「革命年代的字型：俪宋」

分类

上海活字

书刊

亚洲

人物

会刊

公共

All content copyright by The Type (Type is Beautiful) and its authors, unless otherwise noted. Information
除转载内容和特别标注外，所有文字内容版权由 The Type (Type is Beautiful) 和其作者所有。版权信息
请勿转载本站文章。请参阅：使用条款 | 隐私声明与 Cookies

